IN THE CLAIMS:

Please amend the claims as follows:

1. (Original) A method of managing cache in a shared memory multiple processor computer system, comprising:

executing, by a processor, a cache purge instruction that configures the processor to purge a cache line from the processor and send the cache line to at least one of a plurality of processors in the shared memory multiple processor computer system to update the at least one of a plurality of processors.

2. (Original) The method of claim 1, wherein the step of executing, by a processor, a cache purge instruction is performed after modifying the cache line by the processor.

3. (Original) The method of claim 1, wherein the cache line has a unique address.

4. (Original) The method of claim 1, wherein the cache purge instruction updates all processors in the computer system.

5. (Original) The method of claim 1, wherein the cache purge instruction updates only an oldest cache line.

6. (Original) The method of claim 1, wherein the cache purge instruction updates at least one level of cache.

7-11. (Cancelled)

12. (Original) A computer system, comprising a shared memory and at least two processors wherein each processor is associated with at least one level of cache and wherein each processor, when executing a cache purge instruction is configured to:

purge a cache line from the processor executing the cache purge instruction and send the cache line to at least one other processor in the computer system to update the at least one other processor.

13. (Original) The system of claim 12, wherein the processor is further configured to execute the cache purge instruction after the processor modifies the cache line.

14. (Original) The system of claim 12, wherein the cache purge instruction has a unique operations code.

15. (Currently Amended) The system of claim 12, wherein the cache purge instruction is referenced to at least five ~~field~~ fields.

16. (Currently Amended) The system of claim 15, wherein one of the at least five ~~field~~ fields indicates which processors will be updated by the cache purge instruction.

17. (Original) The system of claim 15, wherein one of the at least five fields indicates which level of cache will be updated by the cache purge instruction.

18. (Cancelled)

19. (Currently Amended) A tangible signal bearing medium comprising a program which, when executed by a processor in a shared memory multiple processor computer system performs an operation for managing cache, the operation comprising:

executing a cache purge instruction that will purge a cache line from the processor and send the cache line to at least one of a plurality of processors in the computer system to update the at least one of a plurality of processors.

20. (Original) The signal bearing medium of claim 19, wherein executing a cache purge instruction is after the processor modifies the cache line.

21. (Original) The signal bearing medium of claim 19, wherein the cache line has a unique address.

22. (Original) The signal bearing medium of claim 19, wherein the cache purge instruction updates all processors in the computer system.

23. (Original) The signal bearing medium of claim 19, wherein the cache purge instruction updates only the oldest cache line.

24. (Original) The signal bearing medium of claim 19, wherein the cache purge instruction updates at least one level of cache.

25-28. (Cancelled)

29. (Previously Presented) A method of managing cache in a shared memory multiple processor computer system, comprising:

executing, by a processor, a cache purge instruction that configures the processor to purge a cache line from the processor and send the cache line to at least one of a plurality of processors in the shared memory multiple processor computer system to update the at least one of a plurality of processors, wherein the cache purge instruction updates all caches in the computer system and marks a state of all updated cache lines as shared.

30. (Previously Presented) A method of managing cache in a shared memory multiple processor computer system, comprising:

executing, by a processor, a cache purge instruction that configures the processor to purge a cache line from the processor and send the cache line to at least one of a plurality of processors in the shared memory multiple processor computer system to update the at least one of a plurality of processors, wherein the cache purge instruction updates all caches in the computer system and marks a state of all updated cache lines as temporarily invalid.

31. (Previously Presented) A method of managing cache in a shared memory multiple processor computer system, comprising:

executing, by a processor, a cache purge instruction that configures the processor to purge a cache line from the processor and send the cache line to at least one of a plurality of processors in the shared memory multiple processor computer system to update the at least one of a plurality of processors, wherein the cache purge instruction updates only one cache at a designated processor of the plurality of processors then marks a state of the cache line updated as exclusive at the designated processor and marks a state of the cache line as temporarily invalid at the processor executing the instruction.

32. (Previously Presented) The method of claim 31, wherein a cache line marked as temporarily invalid may not be updated.

33. (Previously Presented) The method of claim 32, wherein a cache line is marked as temporarily invalid because another processor in the computer system requested the cache line marked as exclusive.

34. (Currently Amended) A computer system, comprising a shared memory and at least two processors wherein each processor is associated with at least one level of

cache and wherein each processor, when executing a cache purge instruction is configured to:

purge a cache line from the processor executing the cache purge instruction and send the cache line to at least one other processor in the computer system to update the at least one other processor, the cache purge instruction is referenced to at least five ~~field~~ fields and one of the at least five fields indicates how the state of the updated cache(s) will be marked.

35. (Currently Amended) A tangible signal bearing medium comprising a program which, when executed by a processor in a shared memory multiple processor computer system performs an operation for managing cache, the operation comprising:

executing a cache purge instruction that will purge a cache line from the processor and send the cache line to at least one of a plurality of processors in the computer system to update the at least one of a plurality of processors, wherein the cache purge instruction updates all caches and marks the state of the cache lines updated as shared.

36. (Currently Amended) A tangible signal bearing medium comprising a program which, when executed by a processor in a shared memory multiple processor computer system performs an operation for managing cache, the operation comprising:

executing a cache purge instruction that will purge a cache line from the processor and send the cache line to at least one of a plurality of processors in the computer system to update the at least one of a plurality of processors, wherein the cache purge instruction updates only one cache at a designated processor of the plurality of processors then marks a state of the cache line updated as exclusive at the designated processor and marks a state of the cache line as temporarily invalid at the processor executing the instruction.

37. (Previously Presented) The signal bearing medium of claim 36, wherein a cache line marked as temporarily invalid may not be updated.

38. (Previously Presented) The signal bearing medium of claim 36, wherein a cache line is marked as temporarily invalid because another processor in the computer system requested the cache line marked as exclusive.